INTEGRITY AND PRIVACY IN AUTHENTICATED KEY EXCHANGE PROTOCOL FOR PARALLEL NETWORK FILE

Mane Nitin M. [1] | Chakranarayan Pravin D. [1] | Kerekar Madhuri [1]

[1] Dept. Of Computer Engineering, G.S.M.C.O.E, University of Pune, Maharashtra, India.

ABSTRACT

The problem of key establishment for secure many-to-many intercommunications is inspired because of proliferation/spread of large distributed file systems supporting parallel access to several storage devices. The current Internet standard is main focusing area for such file systems, which use Kerberos to establish parallel session keys between clients and storage devices. After reviewing of the existing Kerberos-based protocol shows that it has a number of limitations: (i) a metadata server providing key exchange between the clients and the storage devises (ii) the protocol does not provide forward secrecy for communication (iii) the metadata server generates, all the session keys that are used between the storage devices and clients and this inherently leads to key escrow. Key exchange protocols that are propose to address above mentioned/described issues. We show that this protocol is capable of reducing up to approximately more than 50% of the workload of the metadata server and concurrently supporting escrow-freeness and Forward secrecy. This requires only a small fraction of raised computation overhead at the user.

KEYWORDS: key escrow, network file systems, authenticated key exchange, Parallel sessions, forward secrecy

I. INTRODUCTION

In a parallel application the file data is spread among the various nodes or devices for giving the concurrent access to multiple tasks using parallel file system. This is widely used in large scale cluster computing which is mainly depends upon the reliable fetch as well as high performance to the large amount of datasets. Due to this the bandwidth of I/O is highly achieved by concurrent data fetching with different number of devices in between the maximum number of clusters which are used for computing. During this the loss of data is prohibited or secured using the data mirroring and defect tolerance striping algorithms are used for data mirroring .There are some examples of highly performance parallel file systems which undergoes in production that uses the IBM General Parallel File Systems.

In this paper we are going to proposed system which describes the way to which keys materials are exchanged as well as how to establish the secure parallel session in between storage devices and clients in (pNFS) Parallel Network File System as well as this proposed system is maintain current internet standard in scalable and efficient way. Particularly, we are trying to meet the following requirements, are not achievable by current Kerberos-based solution or which have not been satisfactorily achieved. The pNFS is introduced by theUMich/CITI, IBM, ENC, and Sun and because of this pNFS has many common features and pNFS is also highly compatible with many of the Commercial Network File System. We are mainly focusing on maintain Integrity and privacy in Authenticated keyexchange protocol for Parallel Network File System.

II. ORGANIZATION

The paper is organized as follows: Related work is presented in Section III. We present our scheme in Section VI. Mathmatical approach in section VIII.The Future works in Section XII. We conclude in Section XIII.

III. RELATED WORK

We have given the first preference to security of more scalable distributed file system. We have seen example in the base paper employed Kerberos for enforcing access control and performing authentication. Symmetric Key techniques are probably used in Kerberos for recent Classification, were best spared distributed environment.

Ocean Store, LegionFS and FARSITE, are the file system and data framework, were built to use of public key infrastructure and public key cryptographic techniques to achieve cross-domain client authentication separately. Public key cryptographic techniques, was established to start for lots ofelectrical grid are classified key management schemes. Every client of this system is expecting to authorize pair of private/public key. Actually the system weren't produce basically for parallel access and scalability performance. With the rising classification of highly spread and network connected storage device system, sub serial work, for the main proposed on scalable security. These proposals produces that a metadata server provides secret key with each storage device. For the authentication group key used to generate message authentication codes. The compromise of the storage device or metadata server gives permission the attacker to act as the server to anybody in entity set of file system. That may be concluded that each storage device shares a many secret key with the main server i.e. metadata server. This method having some restriction for the capability to authorizing input/output on separate for one device, multiple storage devices typically different for the different group of objects or block. Many of the recently proposals that can be accept the hybrid asymmetric key and symmetric key methods, that gives permission to extend any number of storage system, which can be provide excellence security efficiency ratio. Client must be re-use a shared and establish key with a storage device are permitted by the authentication key establishment protocol. Mainly other recent proposals and mata don't together with accurate security analysis.

IV. EXISTING SYSTEM

Parallel Network File System (pNFS), which uses the Kerberos system toestablish session keys between clients and storage devices for data exchange. Our analysis of the existing Kerberos-based protocol shows thatit has a number of limitations.

V. DRAWBACK'S OF EXISTING SYSTEM

- Protocol does not provide forward secrecy.
- Storing Devices have heavy workload that restricts the scalability ofthe protocol.
- Metadata server generates itself all the session keys that are usedbetween the clients and storage devices for data exchange and thisinherently leads to key escrow.

VI. PROPOSED SYSTEM

In this project, we propose Integrity and Privacyfor parallel NetworkProtocol File System in Authenticated KeyExchange. When the dataexchange between clients and storage device starts, at that time we willcheck integrity of the data after fixed amount of time, by using Merkle hashTress algorithm.

VII. MARKEL HASH TREE

Merkle tree is a tree, which contain every non-leaf node is labeled with the hash of the labels or values of its all child nodes. Hash trees are important because they allow efficient and secure conformation of the contents of huge data structures. Hash trees are a generalization hash chains. Important facts regarding Merkle hash Tree Signature Scheme

Security of this signature scheme depends on the security of the hash functions of all its child nodes.

Only one hash needs to be maintained/shared securely and pass/transferred along with data.

To authenticate any data block only log2 n hashes need to be transferred along with data, here n denotes total calculated number of data blocks.

On these term integrity checking of a continuous range of blocks, only one single hash needs to be transferred.

VIII. MATHEMATICAL BACKGROUND

Naming Convention

SD= Storing Device

Ct= convert

Ms= Valid time/session time for data transfer

```
Ds= Data send from storage device

Dc= Data receive at client side

V = Valid time/session time

Hs= hash(Ds)t    if     (0<t<v)

Else    0

Hc= hash (Dc)t   if (0<t<v)

If  Hs== Hc =continue time data transfer

Else Close data transfer

Request for another/ new session

Public int hash(string S)

{

            Int hashcode= 0;

            Int MOD= 10007;

            Int shift= 29;

For(int i=0;i<S.length;i++)

 {

Hashcode= ((shift*hashcode )*MOD+S.charAt(i)*MOD);

}

return hashcode;

}
```

IX. ARCHITECTURE OF PROPOSED SYSTEM

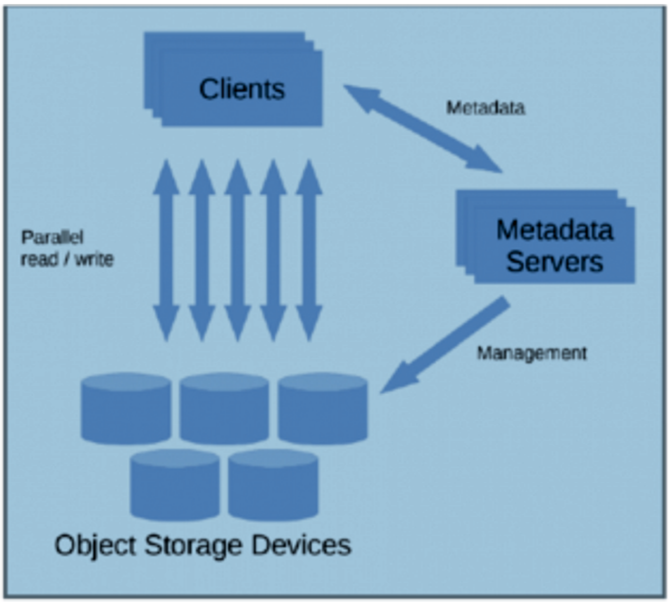

Fig. 1 Proposed Architecture

X. MARKEL HASH TREE ALGORITHM

As described in below flow diagram, suppose a data exchange between client and storage device is in progress. The session is established for 4hour for data exchange, as described in propose system after fixed amount of time like after every 30 min we will check the integrity of the data. After 30 min when storage device send data to client it sends data along with its hash value, at client side client will generate and compare hash value of thedata. If hash matches with the received hash from storage device, client will continue the data transfer otherwise it will generate stop data transfer, because of which integrity and privacy is achieved.

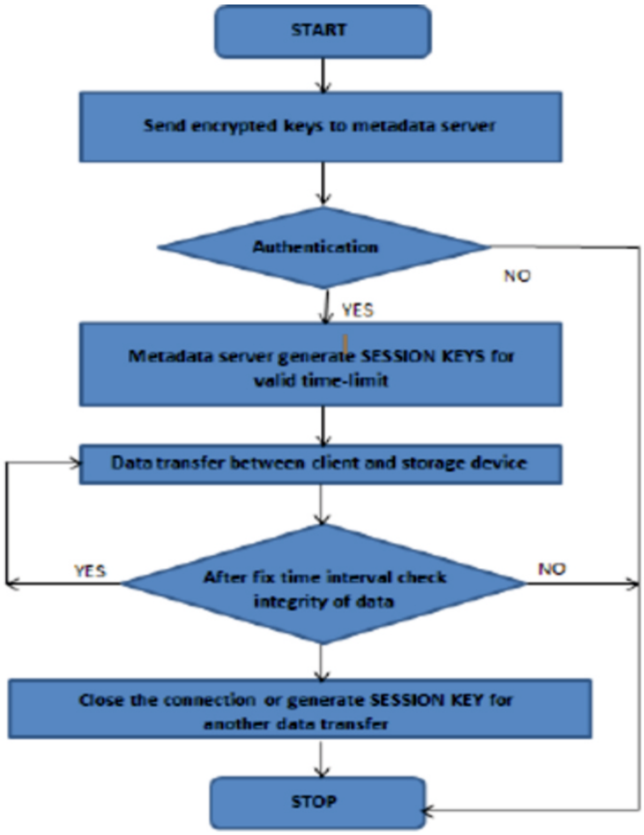

Fig. 2 Flow Of Markel Hash Tree Algorithm

XI. ADVANTAGES OF PROPOSED SYSTEM

- Secure and trustedauthorize key exchange protocols.
- Productivity and security.

XII. FUTURE WORK

We describe a limitation of this approach and our future work as follows

- Require space to store hash of the data
- Slight decrease in process, as time require to find out hash value

XIII. CONCLUSION

In this paper we introduced Integrity and Secure authenticated key exchange protocols for parallel network file system (pNFS). The existing Kerberos-based pNFS protocol has some disadvantages which are overcome in our proposed protocol as well as our protocols gives three Appealing advantages also. First One is our protocol provide secrecy along with escrow-free. Second advantage over Kerberos-based pNFS protocol isthe metadata server which is used in execution having very less workload than the existing Kerberos-based approach. And the last advantage is that our protocol provides basically two forward secrecy: one is fully forward secure (this is respected with the session). While this is the partially forward Secure (this is related to multiple sessions within a specific time period.)

XIV. ACKNOWLEDGEMENT

To study this paper, I would like to be very thankful to my project guide & Coordinator Prof. Shrinivas and Head of the Department Prof Ratnaraj in Computer Department of Genba Sopanrao Moze College Of Engineering related to Savitribai Phule Pune University. We are alsothankful the whole IEEE organization who helps appropriate to search various research papers related to my research. Because of their support only I am able to complete my research note.

REFRENCES

1. C. Adams. The simple public-key GSS-API mechanism (SPKM). *The Internet Engineering Task Force* (IETF), RFC 2025, Oct 1996.
2. A. Adya, W.J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J.R. Douceur, J. Howell, J.R. Lorch, M. Theimer, and R. Wattenhofer. FARSITE: Federated, available, and reliable storage for an incompletely trusted environment. In *Proceedings of the 5th Symposium on Operating System Design and Implementation (OSDI)*. USENIX Association, Dec 2002.
3. M.K. Aguilera, M. Ji, M. Lillibridge, J. MacCormick, E. Oertli, D.G. Andersen, M. Burrows, T. Mann, and C.A. Thekkath. Blocklevel security for network-attached disks. *In Proceedings of the 2nd International Conference on File and Storage Technologies (FAST).* USENIX Association, Mar 2003.
4. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A view of cloud computing. *Communications of the ACM,* 53(4):50–58. ACM Press, Apr 2010.
5. Mr.Shirsath Kirankumar Vilas(Computer Engineer), Author of *"SELF ADDPTIVE SYMANTIC FOCUSED CROWLER FOR INFORMATION DISCOVERY AND DATA MINING"*